Journal of Combinatorial Mathematics
and Combinatorial Computing
www.combinatorialpress.com/jcmcc

Decision Tree Algorithm Based Legal Liability Determination and Contract Fulfillment Path in the Execution of Smart Contracts

Bihua Ou[1], Baomin Wang[1,✉]

[1] *Law School, Xi'an Jiaotong University, Xi'an, Shaanxi, 710049, China*

ABSTRACT

The ontological issues such as the concept, features, and attributes of smart contracts written in code and running on the blockchain have been the focus of research in the academic community. In this paper, we first construct a smart contract illegal behavior determination model based on the C4.5 decision tree algorithm, which realizes accurate prediction and determination of illegal behaviors existing in smart contract transactions by extracting multiple attribute features of smart contract transaction data. Then, the correlation between smart contract features and contract risk is analyzed by Pearson coefficient, and the risk assessment evaluation system of smart contract performance is constructed by using hierarchical analysis. Finally, the fulfillment path of smart contract is proposed by synthesizing all the analysis results. Among the 24 randomly selected samples, the total prediction probability of the illegal behavior determination model based on the C4.5 decision tree algorithm reaches 95.83%, which is able to effectively identify the illegal behavior of smart contracts. The Pearson chi-square value between smart contract features and contract risk is 224.6317, and the Sig.(two-tailed) value is 0.000, indicating that there is a significant correlation between the two. By constructing a smart contract risk assessment index system, this paper designs a dynamic monitoring model of smart contract fulfillment risk level, and proposes a smart contract fulfillment path from the aspects of reasonable allocation of legal responsibility and legal regulation of contract fulfillment.

Keywords: C4.5 Decision Tree Algorithm, Pearson Correlation, Hierarchical Analysis, Smart Contract, Legal Liability Determination

1. Introduction

Early on smart contracts were defined as a computerized transaction protocol that enforces the terms of a contract. The overall goals of smart contract design are to satisfy generic contract conditions (e.g., payment terms, liens, confidentiality, and even enforcement), minimize the occurrence of human-

✉ Corresponding author.
E-mail address: baominwang@mail.xjtu.edu.cn (B. Wang).
Received 01 February 2024; Revised 20 April 2024; Accepted 10 November 2024; Published Online 15 April 2025.
DOI: 10.61091/jcmcc127a-475

caused bad faith and surprises in the contract, and minimize the need for trusted intermediaries. Related economic goals include reducing fraud losses, arbitration and enforcement costs, and other transaction costs [1-5]. Blockchain technology, which is technically based on cryptography and computer principles, belongs to the underlying technological framework that has a generalized nature and continuously makes the intersection of the virtual world and the real world more closely connected. With the rapid development of blockchain technology, blockchain technology began to be combined with smart contracts. Due to the adaptability of blockchain technology as the underlying technical architecture of smart contracts, smart contracts are redefined by blockchain technology and begin to flourish in various fields of society [6-8]. Smart contracts under blockchain technology are of great significance to the innovation of the existing social order and pattern, and their characteristics of "decentralization", "trustlessness", and "autonomy" show great potential for improving economic efficiency, reshaping trust mechanisms, and innovating social governance models [9-10]. Blockchain as the cornerstone of digital economy can effectively empower the digital economy. The main application areas of blockchain-enabled digital economy are supply chain finance, industrial internet, cross-border trade, vehicle networking, and commodity anti-counterfeiting traceability [11-15]. Blockchain can also regulate the digital economy and provide services for digital economy products and technological innovation. Smart contracts based on decision tree algorithms and blockchain technology are applied in the field of civil transactions, which is conducive to the realization of the deep integration of the digital economy and the real economy, and further improves the transaction efficiency, reduces the transaction costs, and maintains the transaction security. Smart contracts are expected to deeply change the traditional business model and social production relations based on their characteristics of being able to establish trust between unfamiliar nodes, being programmable, and not being tamperable, laying the foundation for the construction of programmable assets, systems, and societies [16-20]. At present, the research on smart contracts in various countries is not yet in-depth, in order to better utilize the value of smart contracts and serve the development of economy and society, the proposed decision tree algorithm based on decision tree algorithms for determining the legal responsibility and contract fulfillment paths in the process of smart contract execution can put forward suggestions to solve the legal problems arising in the process of smart contract execution, which has an important theoretical research value and practical significance [21-25].

In this paper, an illegal behavior prediction model is constructed based on the C4.5 decision tree algorithm, which realizes the effective determination of illegal behavior in smart contracts. Then, it explores the correlation between smart contract features and contract risk through Peasron correlation analysis method, and realizes the evaluation index system of smart contract performance risk by using hierarchical analysis method and grey correlation analysis method, and designs the dynamic detection mechanism of smart contract performance risk to cope with the risk of smart contract and ensure the effective performance of smart contract. Finally, the specific path of smart contract fulfillment is designed from the aspects of responsibility allocation and legal rules.

2. Determination of legal liability in smart contracts

2.1. Blockchain and Smart Contracts

2.1.1. Blockchain technology. The idea of blockchain technology first originated in Bitcoin and became publicized as the underlying technology for Bitcoin, essentially making blockchain technology superior as a new combination of technologies in a distributed environment. The key technologies included are cryptography, P2P networks, digital signatures, timestamps, consensus algorithms, and smart contracts, among other legacy technologies. Distributed, on the other hand, means that all data is not dependent on a specific data server, but is stored as a ledger-like form in each node participating in the blockchain network. The blockchain is like a linked train car, which serves as a form of data storage for blockchain technology, linking one block after another in the chronological order of block

generation, with each block recording the corresponding information. The structure of the data block is shown in Figure 1, which contains two parts: the block header and the block body.

The version number of the current block, the pointer to the address of the previous block, the timestamp, the random number, the target hash value of the current block, the root plant of the Merkle tree and other information are contained in the block header. Among them, Merkle hash tree is a kind of tree in the data structure, which is a class of binary or multinomial trees based on hash value, with all the characteristics of tree structure. The hash value of its data block is stored as the value of a leaf node in a Merkle tree, while the value on a non-leaf node is the hash value that combines the results of all the children of that node. The Merkle tree in the blockchain is a binary tree, which is used to store transaction data information.

Transaction information is mainly recorded in the block body. Each transaction is permanently stored in the data block and linked to the blockchain, and the information is available to every node in that blockchain network. To ensure that each transaction is not altered, forged, or duplicated in the chain, a Merkle tree located in the block body is used to sign each transaction using data signature technology, and a hash process in the Merkle tree generates a unique Merkle root value for all transactions and records it in the block header. All the blocks are connected in chronological order to form the blockchain.

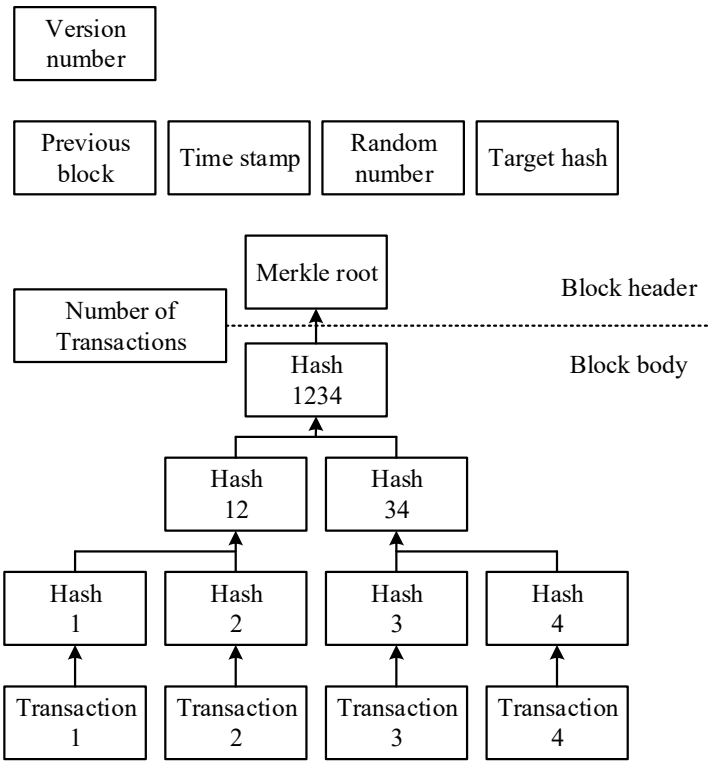

Fig. 1. Block structure

Overall, blockchain has the characteristics of distributed (or decentralized), time-series data, collective maintenance, programmable, and secure and trustworthy, which makes blockchain technology can be widely used in digital currency, financial industry, and social systems. First, the distributed nature of blockchain can circumvent the high cost and risk of centralization as well as the problems of malicious node attacks. Secondly, the collective maintenance and time-sequential data of blockchain technology make the data tamper-proof and traceable, which can guarantee the authenticity of transactions. Finally, smart contracts in blockchain greatly improve efficiency.

While gaining the advantages of decentralization, non-tampering and traceability, blockchain loses the advantages of traditional systems that can save storage space and high efficiency. Therefore, the

application of blockchain technology must be weighed against application benefits and efficiency. In small-scale applications with low frequency of information exchange and low total amount of information, the cost waste and inefficiency caused by the use of blockchain technology is not obvious, and conversely, the defects of high cost and low efficiency are particularly obvious in large-scale applications.

2.1.2. Smart Contracts and Their Typical Applications. A smart contract is a set of digitally defined commitments, including protocols on which contract participants can execute those commitments, which, once developed and deployed successfully, can be self-executing and verifiable without outside human interference.

Smart contracts have the following five advantages in general:

1) Decentralization: In general, smart contracts do not require a centralized third party to arbitrate whether the contract is executed smoothly according to the provisions, and the supervision and arbitration of smart contracts are completed by computers.

2) Low-risk human intervention: after a smart contract is deployed, nothing in the contract is allowed to be modified, and no party in the contract will intervene in the execution of the contract, i.e., once the contract is deployed, any party involved in the contract cannot destroy it at will. Even if the contract is destroyed, the party responsible for the destruction event will have to bear the corresponding penalty, and this penalty is also determined when the contract is formulated, and cannot be changed after the contract is deployed and takes effect.

3) Efficient execution accuracy: Since the terms of the smart contract and its execution process are formulated in advance and executed under the absolute control of the computer, all the execution results of the smart contract are accurate, i.e., there will be no unforeseen results.

4) Lower operating costs: it is because smart contracts have the characteristic of de-humanization intervention, which greatly reduces the cost of fulfilling the contract, adjudicating the contract and enforcing the contract.

5) Efficient real-time updates: Since the execution of smart contracts does not require human intervention, i.e., it does not require the involvement of an authoritative third party or a centralized proxy service, it can respond to customer requests at any time, which greatly enhances the efficiency of transaction implementation.

These advantages of smart contracts are obvious, but their widespread use faces unknown risks. If there is a vulnerability in the design of a smart contract, its de-humanizing nature can make it impossible to fix it online.

Ether is a new open blockchain platform with smart contract functionality and a built-in Turing-complete programming language that allows anyone to build and use contracts and decentralized applications running on blockchain technology, and to set up their own freely-defined rules of ownership, transaction methods, and state transition functions. Accounts, the core operating objects of Ether, are categorized into two types: externally owned accounts and contract accounts. External accounts are ordinary accounts encrypted and shared by a public-key cryptosystem with access to cryptocurrency and no associated code. Contract accounts are controlled by contract code and deploy smart contracts.

Users in the ethereum blockchain system can create smart contracts by posting a transaction. Smart contracts can store and forward value to other accounts or contracts, and once triggered, smart contracts execute automatically without interference. Only when an external account issues a command, the contract account will perform the corresponding operation, so the contract account cannot spontaneously perform operations such as arbitrary digital generation or application program interface invocation, and will only do these operations when called by an external account. That is, the external account sends a message to other external account or contract account by creating and signing a transaction. The message between two external accounts is just a value transfer, but the

message sent by the external account to the contract account will activate the code inside the contract, thus executing various operations, and the contract account can only receive the transaction sent by the external account, and it cannot start a new transaction.

2.2. Legal Liability in the Execution of Smart Contracts

2.2.1. Types of responsibilities during smart contract execution. The creation and execution of a smart contract is essentially a contractual act and should be subject to the provisions of contractual liability, which refers not only to the liability arising from the breach of contractual obligations, but also to the liability arising from the breach of statutory obligations under the Civil Code. According to the overall "life flow" of the contract from negotiation, establishment, entry into force, fulfillment and termination, the contractual liability can be generally divided into four types, namely, contractual negligence, anticipatory breach of contract, breach of contract and post-contractual liability.

There will be no subjective fault in the execution of smart contracts, i.e., all the constituent elements cannot be fulfilled at the same time, and there will be no negligent contracting and post-contractual liabilities, so there will be only two possibilities, i.e. anticipatory breach of contract and breach of contract. Anticipatory liability, refers to the expiration of the contract, one party expressly or by their own behavior that does not perform the main debt and need to bear the liability for damages. Liability for breach of contract refers to the liability for damages for non-performance of the contract when the deadline for performance expires, either expressly or by the debtor's own conduct. The liability for anticipatory breach of contract and breach of contract follows the principle of no-fault attribution. Therefore, even smart contracts involve both of these liabilities when they execute transactions. However, the above argument only applies when the smart contract party triggers a loss, and it does not mean that it applies to the human parties in the transaction behavior, which should be liable for compensation if the liability for negligence in contracting and post-contractual liability arises due to the human subjects in the transaction relationship.

2.2.2. Liability of Smart Contracts under Blockchain:

1) The subject of liability is unclear. The assumption of legal tort liability should have a clear tort subject, when one party violates the legal or agreed obligations, the other party can assume the breach of contract or tort liability to the tortfeasor or the third party organization. However, because the nodes of the blockchain are not one-to-one relative to the real-life people, it is difficult to confirm the real subject of responsibility. Blockchain smart contract in the whole operation process not only the contractual parties but also the users of other nodes, ethereum blockchain technology developers, financial institutions and so on. When there is a mistake in the conversion of the written contract to the smart treaty code, or the blockchain code runs incorrectly, or the network platform is maliciously attacked, it is difficult to confirm who is the real responsible body of the contract code converter, the blockchain platform provider, and the developer of the technology. Any participant in the blockchain transaction process can adversely affect the contract. Existing laws do not specify the tort liability system, and it is difficult to recognize the tort subject at this stage, so the legitimate interests of the contract subject are not protected, which hinders the development of blockchain smart contract economy.

2) Unclear allocation of responsibility. According to the blockchain smart contract, the contract fulfillment terms are integrated in advance to exclude all uncertainties. Automated and precise execution is carried out by codification to guarantee fair transactions and reduce operating costs. In the setting of the smart contract, there will be no wrong performance, and the subject will not violate the agreed or legal obligations, and will not be liable for violating the obligations. However, nothing is absolutely perfect, because the existing level of technology may still occur when the technicians

transform the smart contract code errors, or by hackers malicious attacks lead to the contract purpose can not be achieved, and even if the specific responsibility can be determined by the subject can not be assigned responsibility, because the existing law in practice does not specify how the subject of the smart contract is liable.

3. Decision Tree Algorithm Based Determination of Illegal Behavior in Smart Contracts

Aiming at the problems of unclear responsibility subject and unclear responsibility allocation in smart contracts, as well as the resulting smart contract scam, this paper constructs an illegal behavior determination model based on the C4.5 decision tree algorithm. By analyzing the smart contract transaction data, extracting various attribute features, analyzing the data indexes under the attribute features, and comparing the model with other models in the same environment, the model can accurately predict the illegal behaviors in the smart contract transactions, so as to realize the determination of legal responsibility in the process of executing the smart contract, and to improve the transaction environment based on the blockchain.

3.1. Classification algorithms

Classification algorithms are concerned with finding a better solution to the classification problem. Classification algorithms are actually trained to find a classification rule from certain specific data sets. After that, this classification rule is then used to recognize a new class of data. The specific process of classification algorithm is as follows:

1) Input training data sets. These training sets consist of one record, each record contains attribute feature vectors and category labels.

2) Classification process. By discovering the attribute feature vectors and the class labeling laws corresponding to them, the corresponding rules are derived.

3) Output classification model. This rule model can be used to predict the class label to which a particular record belongs. Classification algorithms have considerable applications in real-life risk assessment prediction, text retrieval, and so on. Currently, the classification algorithms that have been researched more in academia are decision trees, K nearest neighbors (KNN), Bayesian classification, support vector machines (SVM), and so on. In this paper, WEKA open source software will be used to detect the classification accuracy of machine learning algorithms such as C4.5 Decision Tree, KNN, Plain Bayes, Bayesian Networks, SVM, etc., and ten-fold cross-validation method will be used to experimentally analyze the dataset and compute the algorithm's classification accuracy as shown in Equation (1):

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

where, TP and FP are the instances of correct and incorrect classification respectively.

The classification accuracy of the five machine learning classification algorithms can be obtained from the test results as shown in Table 1. As can be seen from Table 1, the model classification accuracy results of the five common machine learning classification algorithms are similar. However, overall, the accuracy of C4.5 decision tree classification algorithm is slightly higher than the other four, reaching 87.49%. Therefore, this paper chooses it as the method for illegal behavior determination during smart contract execution.

Table 1. Comparison of classification accuracy

Classification algorithm	Accuracy rate
C4.5 decision tree	87.49%
KNN	77.38%
Bayesian networks	85.26%
Naive bayes	83.65%
SVM	85.17%

3.2. C4.5 Decision Tree Algorithm

Decision tree is a very important classifier, which has the advantages of simple learning, high classification accuracy and low domain knowledge requirement. Generally, a "learned" decision tree consists of a root node, several child nodes, and several leaf nodes. The leaf nodes represent the class labels corresponding to the feature vectors of an attribute, the root node contains the set of all attributes in a record, and the child nodes are divided by the attribute tests. The key point of studying the decision tree algorithm is how to choose the optimal method of dividing attributes, which is also the difference between different classification algorithms, such as ID3, C4.5, CART, SLIQ and so on. This paper focuses on the C4.5 decision tree algorithm based on information entropy, C4.5 is improved on the basis of ID3 research [26]. The basic process of C4.5 decision tree algorithm is shown in Fig. 2, which demonstrates that attribute partitioning is the key to build a decision tree, firstly, a segmentation feature that best distinguishes the current sample data is selected as the root node, and then it is judged whether the segmented samples all satisfy the pruning strategy, until the division to no remaining attributes.

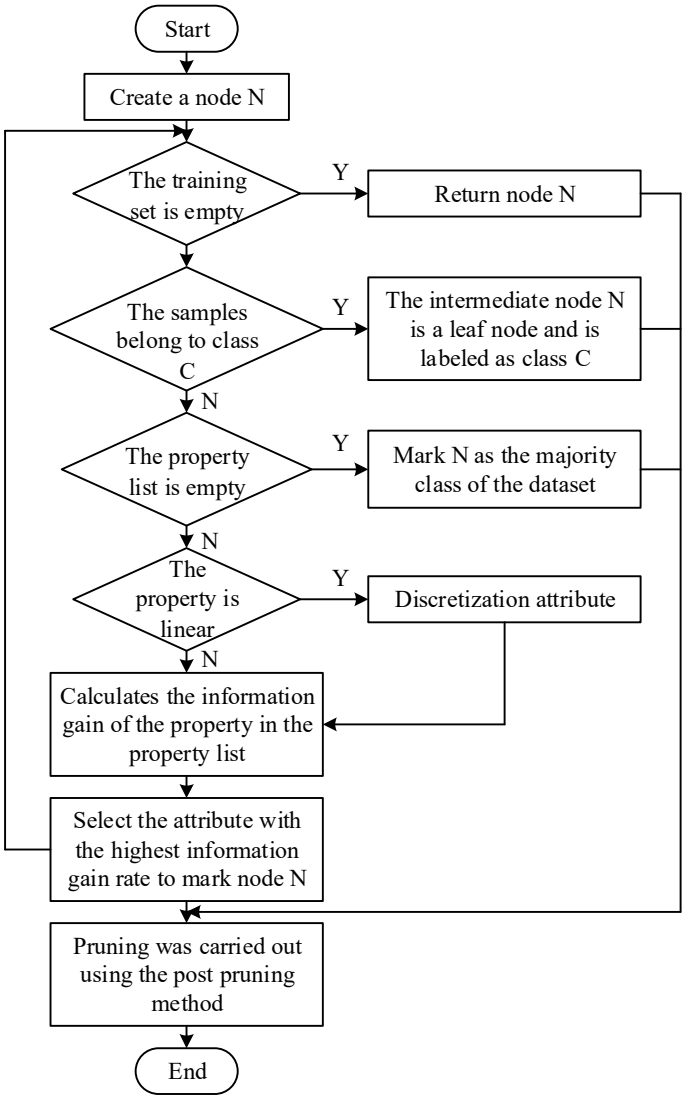

Fig. 2. Basic flow diagram of the C4.5 decision tree algorithm

C4.5 In order to solve the problem of ID3 using information gain to divide the attributes, the problem of selecting more all favorites for the attribute feature values occurs, and the method of optimal division of attributes by gain rate is proposed. Following the C4.5 decision tree algorithm process to determine the illegal behavior of smart contracts in Ethernet, the process is as follows:

1) Calculate the information entropy. It can measure whether the sample sets belong to the same category. It is defined as follows: assuming that the proportion of category k samples in the current training set D is P_k (since the categories of the Ethernet smart contract dataset categorized in this paper are illegal and legal, the value of k here is 1, 2), then the category information entropy is computed based on the training set D as:

$$Info(D) = -\sum_{k=1}^{2} P_k \log_2 P_k \tag{2}$$

2) Calculate the information gain. In the Ethernet smart contract training set D, attribute feature vector a is selected to divide set D, and feature vector a has a total of 10 possible values, then, taking the v th value is selected in the sample set D with value a^r and is noted as D^v. The information entropy of D^v can be calculated from Eq. (2) and weights $\frac{|D^v|}{|D|}$ are assigned to the

branch nodes, which are used to solve the problem of different number of samples contained in the branch nodes. Thus, the information gain of using attribute feature vectors a to partition the set D can be expressed as:

$$Gain(D,a) = Info(D) - \sum_{v=1}^{10} \frac{|D^v|}{|D|} Info(D^v) \tag{3}$$

However, the use of information gain to select the dividing attributes can create a preference for attributes with more selected values. This affects the generalization ability of the decision tree. Therefore, the C4.5 decision tree algorithm selects the gain rate to select the division attributes.

3) Calculate the gain rate.The calculation of gain rate in C4.5 decision tree algorithm is extended on the basis of information gain.The training set D uses the gain rate of attribute feature vector a as shown in Equation (4):

$$Gain_ratio(D,a) = \frac{Gain(D,a)}{IV(a)} \tag{4}$$

Among them:

$$IV(a) = -\sum_{v=1}^{7} \frac{|D^v|}{|D|} \log_2 \frac{|D^v|}{|D|} \tag{5}$$

It is worth noting that the C4.5 decision tree algorithm first uses a heuristic to filter out candidate dividing attributes with information gain greater than the average value in order to prevent the gain rate criterion from falling into the predicament of selecting fewer values for the preferred attributes.

3.3. Experimentation and Analysis

3.3.1. Assessment of model effectiveness. In order to evaluate the performance of the smart contract illegal behavior determination model based on the C4.5 decision tree algorithm, this paper introduces the logarithmic loss function (Logloss), which represents the number of iterations that the model algorithm starts to execute in the training set and the test set, respectively, and the execution of the Logloss in the different datasets is shown in Figure 3.

According to the curve trend situation presented in Fig. 3, it can be clearly observed that the logarithmic loss function (Logloss) gradually tends to stabilize after 120 iterations in the C4.5 decision tree algorithm in the training set and test set respectively, while the loss function in the test set is significantly higher than that in the training set, which further argues that the final model constructed has a better capability of illegal behavior detection.

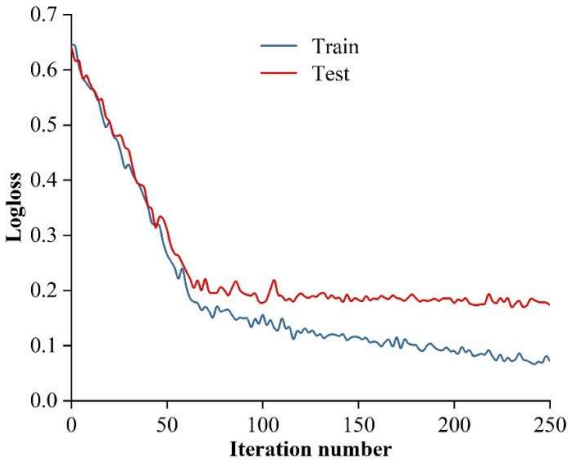

Fig. 3. Change trend of loss function

Data sampling also plays a key role in the assessment of the model, and the selection of data through different sampling ratios can more comprehensively ensure the accuracy and applicable performance of the model. Therefore, in this paper, in order to facilitate the assessment of the performance of the model, 24 samples were first randomly selected from the experimental data to observe whether these samples reach the consistency with the authenticity of the original data under the treatment of the constructed model, and the model prediction results of these 24 samples are shown in Table 2. Among them, label 1 indicates that it belongs to illegal behavior, and label 0 indicates that it does not belong to illegal behavior.

According to Table 2, it can be seen that in the 24 randomly selected samples, the comparison between the final measurement results and the real results is basically the same, and its final total prediction probability reaches 95.83%, which indicates that the constructed model has a strong detection ability for these 24 randomly selected samples. However, the sample number is too low to reflect the truthfulness of the model, so it is necessary to expand the sample capacity and make further determination of the constructed model.

Table 2. Model prediction results of 20 samples randomly

Sample location	Sampled labels	Label for model predictions	Sample location	Sampled labels	Label for model predictions
3174	1	1	2706	1	1
1428	0	0	4085	0	0
3185	0	0	1041	1	1
457	1	1	4023	0	0
306	1	1	236	1	1
304	0	0	393	1	1
1832	1	0	4065	1	1
4398	0	0	2847	1	1
1159	0	0	1124	0	0
2534	1	1	558	0	0
3027	0	0	814	1	1
3571	1	1	3126	1	1
Total forecast probability	95.83%				

The feedback of the model prediction accuracy (ACC) values was carried out by gradually expanding the sample capacity, and the ACC change curve is shown in Figure 4.

Figure 4 reflects that the model prediction probability from the initial extraction of a small sample size presents a potent high level, and then with the gradual expansion of the sample size presents a substantial decline, which is due to the construction of the model in further adapting to the sample content of the model and ultimately stabilized, and according to the continuous expansion of the sample capacity until reaching the full dataset used in this experiment, the model's final ACC value reaches 93.89%, and the higher ACC value reflects that the model in this paper has a high predictive ability and applicability to the illegal trading behavior during the execution of smart contracts.

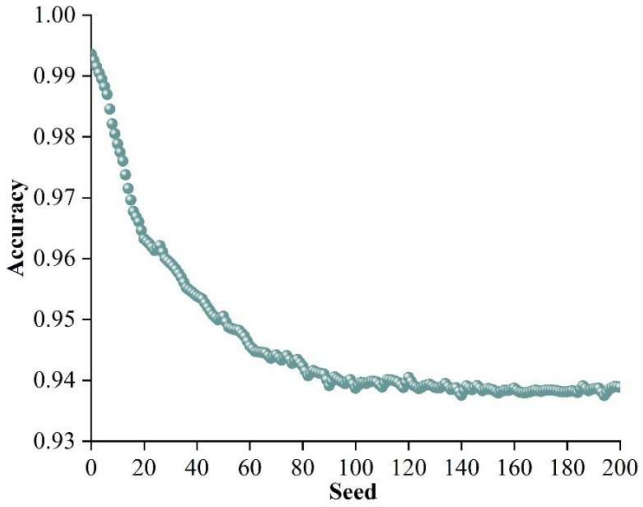

Fig. 4. Prediction rate change for the random sampling range

3.3.2. Characteristic importance. The attribute characteristics of the data have a substantial impact on the final construction of the model, and for the calculation method of decision tree type, the branch decision of the tree needs to be made according to the feature attributes in the process of building the model, so the importance of the feature attributes can be further analyzed to further analyze the adaptability of the model, and the data set structure can be adjusted according to the analysis results or a new attribute feature category can be further created to improve the performance and accuracy of the model. According to the experimental datasets selected in this chapter, the importance of the model feature attributes is ranked, and the top ten important characteristic features are listed according to their importance, as shown in Figure 5.

From the results in Figure 5, it can be seen that the importance value of the feature attribute "Total_ether_balance" is as high as 26.94%, which is much higher than that of other attributes, so it is very important for the final construction of the model, and the specific influence of the feature of this attribute should be considered in the subsequent experimental exploration. The importance of the three attribute features of "Avg_min_between_sent_tnx", "Unique_ Received_From_Address" and "Time_Diff_between_first_and_last_ (Mins)" is almost at the same level, so it is possible to find relationships with these three attributes to prepare for the construction of new attribute features. The six predictors of "Avg_val_sent", "Unique_Sent_To_ Addresses", "Min_value_received", "Total_transactions_(including_tnx_to_create_contract", "Max_val_sent", and "Min_val_sent" showed a lower level of importanceHowever, the influence on the model construction still plays a key role, and most of the importance values of the subsequent 10 attribute features are at a low level or even reach a value of 0.

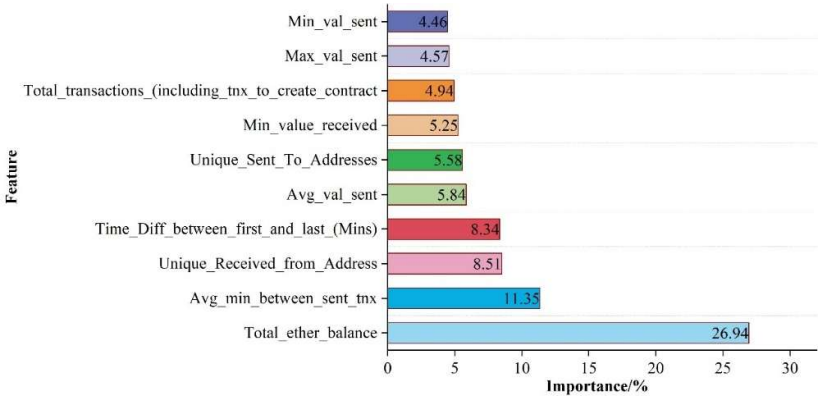

Fig. 5. Top 10 predictors of importance

The importance of the complete predictor variables is shown in Table 3. According to the effect of attribute features on the final construction of the model reflected in Table 3, it can be seen that in addition to the top ten important attribute features listed before, the remaining attribute features focusing on the newly created attribute feature “Sent_Diff_between_max_and_min”, i.e., the difference of the ethereum transactions, ranked 14th in the construction of the model, which also proves that the newly constructed attribute feature has a certain degree of influence on the performance of the model. The attribute “Sent_Diff_between_max_and_min”, i.e., the difference between Ether transactions, is ranked 14th in the construction of the model, which proves that the newly created attribute feature has a certain degree of influence on the performance of the model. It can be seen that different attribute features have different degrees of influence on the model, so in the pre-preparation work for model construction, it is necessary to do more about the characterization process of the data in order to find out the different dimensional differences of the data, in order to further improve the applicability and accuracy of the final construction of the model, and to achieve the experimental purpose that the model needs to accomplish.

Table 3. The importance of the final predictor variable

Serial number	Attribute name	Importance ranking	Degree of impact
1	Avg_min_between_sent_tnx	6	5.95%
2	Avg_min_between_received_tnx	9	3.08%
3	Time_Diff_between_frst_and_last_(Mins)	4	7.11%
4	Sent_tnx	2	10.08%
5	Unique_Received_From_Addresses	3	8.51%
6	Unique_Sent_To_Addresses	5	6.17%
7	Min_value_received	8	5.25%
8	Avg_val_received	13	2.31%
9	Min_val_sent	11	2.45%
10	Max_val_sent	10	2.57%
11	Sent_Diff_between_max_and_min	14	2.10%
12	Avg_val_sent	7	5.84%
13	Total_transactions	17	1.60%
14	Total_ether_received	12	2.39%
15	Total_ether_balance	1	26.94%
16	Total_ERC20_tnxs	16	1.62%
17	ERC20_total_Ether_received	18	1.54%
18	ERC20_uniq_sent_addr	15	1.76%
19	ERC20_uniq_rec_addr	20	1.28%
20	ERC20_max_valrec	19	1.45%

In order to more comprehensively observe the degree of influence of attribute features on the model in the dataset, the SHAP Value factor is especially introduced into the determination factor, and the SHAP Value factor can concretely analyze the positive and negative influences of the attribute features on the construction of the model, which is adjusted through the visualization to obtain the visualization of the positive and negative influences of the top ten important feature attributes as shown in Figure 6.

According to the visualization results in Figure 6, the color represents the size of the feature value and the width represents the feature distribution. However, if the SHAP value is less than zero, it has a negative effect on the label features. Otherwise, it has a positive effect on the label characteristics. In the figure, the positive influencing factors of the characteristic attribute of "Unique_Received_from_Address" are significantly higher than the negative influencing factors, while the negative influencing factors of the two characteristic attributes of "sent-tnx" and "Time_Diff_between_first_and-last-(Mins)" are higher than the positive influencing factors. By

introducing the SHAP Value variable, the positive and negative effects of different attribute features on the model construction can be intuitively determined.

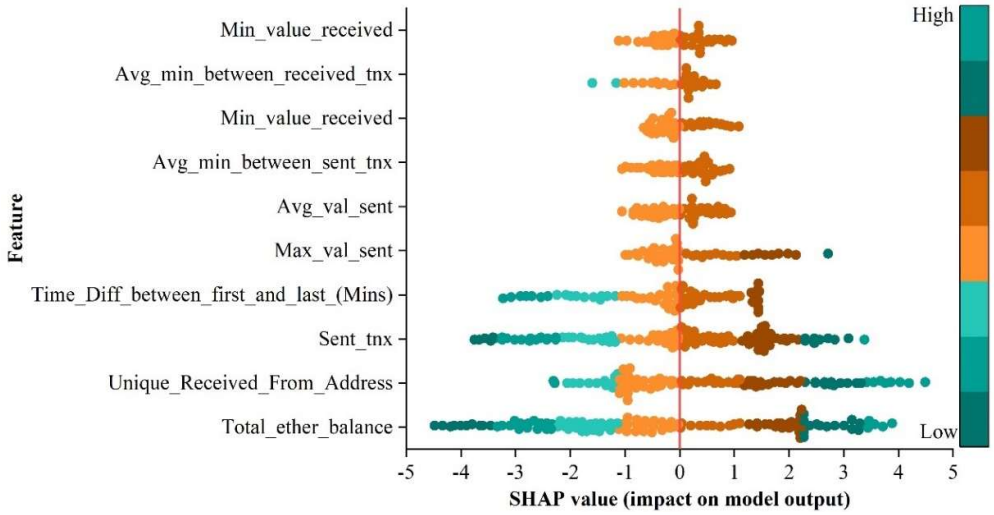

Fig. 6. Visualization of SHAP value

4. Contractual fulfillment paths during the execution of smart contracts

In order to explore the contract fulfillment path during the execution of smart contracts, this paper first analyzes the correlation between smart contract features and contract risk, then explores the assessment method of smart contract fulfillment risk, and finally puts forward a strategic proposal for regulating smart contracts under the blockchain.

4.1. Correlation analysis between smart contract features and contract risk

In order to study the degree of correlation between smart contract features and contract risks, this paper takes 425 Ethernet smart contracts as objects and selects 300 smart contract cases with the highest correlation as samples for quantitative analysis. After classification and screening, 245 typical cases with direct correlation were selected, and then the chi-square test was conducted with the help of SPSS statistical software.

The chi-square test used in this paper is a hypothesis testing method, which is a non-parametric test, and is used to statistically determine the degree of fit between the actual observed values and the theoretical inferred values of two and more samples.

Analyzed by SPSS chi-square test, it can be obtained that the Pearson chi-square value between smart contract characteristics and contract risk is 224.6317, and the Sig.(two-tailed) value is 0.000, which indicates that there is a significant correlation between the two and the asymptotic significance is under 0.001.

Meanwhile, the cross results of smart contract features and contract risk types obtained by SPSS chi-square test are shown in Table 4. Among them, the main features of smart contracts include automatic execution, transparency, non-tamperability, decentralization, and cost-effectiveness, and the contractual risks of smart contracts mainly include the risks of arithmetic attack, code leakage, incorrect execution, privacy protection, and contract regulation.

As can be seen from Table 4, the automatic execution feature and the non-tamperability feature of smart contracts lead to the risk of incorrect execution, which makes the incorrect smart contract cannot be revoked. The transparency feature leads to the risk of code leakage and privacy protection. The decentralization feature and the cost-effectiveness feature lead to the difficulty of regulating smart contracts, which results in regulatory risk. Thus, when formulating smart contracts and designing the

corresponding contract fulfillment paths, it is important to consider them in conjunction with the characteristics of smart contracts to minimize contract risks.

Table 4. Cross results of smart contract influencing factors and contract risk types

Feature	Risk type					Total
	Computational attack	Code leak	Error execution	Privacy protection	Contract supervision	
Automatic execution	0	0	38	0	0	38
Transparency	0	18	0	21	0	39
Immutable	0	0	47	0	0	47
Decentralization	0	0	16	0	31	47
Cost-effectiveness	0	0	0	0	34	34
Total	0	18	101	21	65	205

4.2. Smart Contract Fulfillment Risk Assessment

In order to assess the risk of smart contract in the process of execution, this paper conducts further research on contract fulfillment risk and constructs a contract fulfillment risk evaluation index system based on the hierarchical analysis method [27]. The index system includes three first-level indicators, including quality risk (A), schedule risk (B) and funding risk (C), and is subdivided into production risk (A1), personnel risk (A2), quality management system risk (A3), ancillary risk (A4), technical status risk (A5), test quality control risk (A6), production schedule risk (B1), quality problem risk (B2), cost control risk (C1) and supporting payment risk (C2) 10 secondary indicators.

The weights of contract performance risk evaluation indicators are obtained through the weight calculation formula as shown in Table 5. It can be seen that quality risk accounts for the largest proportion of contract performance risk, with a weight of up to 0.6537, which is the first factor that should be considered. It is followed by schedule risk with a weight of 0.2648, which also needs to be taken into account. And although the funding risk only accounts for 0.0815, it should not be ignored. In addition, from the comprehensive weight of the indicators, the quality management system risk is the most important, accounting for 0.2729, followed by the production plan risk, production risk and supporting risk, which are all greater than 0.10.

Table 5. Contract performance risk evaluation index weight

Primary index	Weight	Secondary index	Weight	Composite weight
A	0.6537	A1	0.2146	0.1403
		A2	0.0293	0.0191
		A3	0.4175	0.2729
		A4	0.1598	0.1045
		A5	0.0714	0.0467
		A6	0.1074	0.0702
B	0.2648	B1	0.72	0.1907
		B2	0.28	0.0741
C	0.0815	C1	0.178	0.0145
		C2	0.822	0.0670

In order to quantify the indicators to assess the performance risk of smart contracts, this paper constructs a quantitative model of contract performance risk evaluation indicators $R = f(F, L, A)$, in which R denotes the performance risk evaluation indicators, frequency F denotes the frequency of occurrence of risk events, degree L denotes the failure impact of risk events, and attention A denotes

the attention paid to the contract at all levels. Frequency F and degree L are obtained by forward backtracking on historical regulatory data, and attention A is obtained by forward analyzing smart contracts.

The comparison sequence is determined according to the $R = f(F, L, A)$ model, and the reference sequence, i.e., the sequence consisting of the worst value of each index is selected, and the performance risk evaluation of contracts A~E is obtained as shown in Table 6.

Table 6. Contract performance risk assessment

Primary index	Weight	Secondary index	Weight	Contract A	Contract B	Contract C	Contract D	Contract E	Worst value
A	0.6537	A1	0.2146	4	4	6	2	2	6
		A2	0.0293	0	0	18%	7%	2%	18%
		A3	0.4175	14	14	14	12	9	14
		A4	0.1598	6	3	3	2	2	6
		A5	0.0714	3	5	2	0	0	5
		A6	0.1074	3	0	4	8	0	8
B	0.2648	B1	0.72	70	100	260	180	0	260
		B2	0.28	0	0	140	80	0	140
C	0.0815	C1	0.178	0	0	3%	6%	0	6%
		C2	0.822	0	0	0	2	0	2

The correlation coefficient was then found to be $\gamma = \begin{bmatrix} 0.664 \\ 0.625 \\ 0.837 \\ 0.628 \\ 0.442 \end{bmatrix}$ based on gray correlation analysis [28].

As a result, contract grading is based on the risk level of contract performance and contracts are categorized into those with a low, medium and high risk level of performance. Different supervisory strategies are adopted for contracts of different grades, with supervisory methods ranging from simplified supervision, normal supervision to more stringent supervision, and the supervisory power invested ranging from less to more.

Among them, Contract C has a high risk level of fulfillment and an enhanced supervision strategy is recommended. Contract A, B, D fulfillment of the risk level of medium, it is recommended to take regular supervision strategy. Contract E fulfillment risk level is low, it is recommended to adopt the simple type supervision strategy. Meanwhile, the risk radar charts of Contracts A, B, C, D and E can be obtained.

By summarizing the above analysis process, this paper proposes a dynamic monitoring model of smart contract performance risk level as shown in Fig. 7, i.e., first take appropriate methods to assess the smart contract performance risk, then classify the contract based on the contract performance analysis, then implement the contract supervision in accordance with the classification supervision strategy, and finally dynamically adjust the smart contract performance risk level, and then continuously recycle, so as to realize the dynamic monitoring of the smart contract performance risk. Dynamic monitoring.

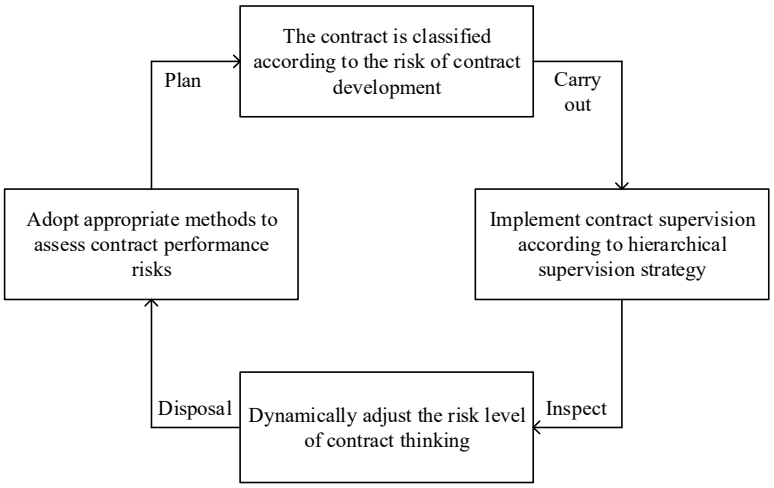

Fig. 7. Dynamic monitoring model of contract performance risk level

4.3. Fulfillment Path Design of Smart Contracts under Blockchain

4.3.1. Rational Allocation of Responsibility for Smart Contracts under Blockchain. In order to ensure the smooth fulfillment of smart contracts, their legal liabilities should be reasonably allocated. For the parties to a blockchain smart contract, both parties to the contract should take various effective measures to ensure the normal fulfillment of the contract, because the cost of cancellation of the smart contract is very high and may exceed the loss brought by the contract. The following situations may occur in the performance of smart contracts: the contracting parties successfully conclude a blockchain smart contract, but in the course of performance, the virtual property cannot be used due to the death of one of the parties, or the platform program malfunctions, resulting in the failure of some of the conditions for the entry into force of the contract, and at this time, it is necessary for the non-performing party to compensate the defending party to protect the damaged reliance interests of the party, and the defending party also obtains the The defaulting party also obtains the right to claim damages from the breaching party, and can claim that the other party bears the responsibility of damages to itself. In this case, the determination of the damages liability of the breaching party is based on the way of assuming damages in traditional contracts. In addition, if any party to the contract concluded in the form of a blockchain smart contract has an untrue expression of meaning, the contract cannot take effect, and if it leads to the loss of the other party or the relevant person, the at-fault party still needs to bear the responsibility for the contracting fault to the injured party, and the matter can then be similarly determined and allocated based on the attribution theory of the traditional contract, and if it causes bodily injury to another person, the victim can also request that he or she bear the Tort liability.

In addition, if such damage is caused by a third party, the third party needs to be held liable for damages. For example, the contracting parties agreed to submit the smart contract to be prepared by a third-party code organization, but because of the third party's mistakes, the contracting parties suffered losses. At this point, it can be determined that the third-party agency did not fulfill the obligation of bona fide contracting, the contracting parties can be held responsible for the responsibility of the agency, as for the characterization of the responsibility, either because of the violation of the contract of entrustment constituted by the breach of contract, or can also cause damage to both sides of the tort liability. It is important to note that the injured party can demand both breach of contract and tort liability from the commissioning party, as well as tort liability from the platform designers and technology developers.

For the technological personnel in the smart contract, the technological personnel of the smart contract, from the establishment and operation of the blockchain Ether platform, the birth of the smart

contract was founded, the operation, maintenance and design of the smart contract plays a technical support role, and plays a vital role in whether the smart contract is successfully performed. Therefore, this paper advocates that contract technicians should be explicitly regulated in the legislation, and specific provisions should be added to the law to monitor their behavior.

As for the code being used for criminal activities, the code can be regarded as a commodity, and the relevant provisions of product defect liability can be applied. Regarding the liability for defects in technical products, if the loophole is caused by the technician's own mistakes, he should be held responsible. If it is caused by a third party, the technician can pursue the actual responsible person after assuming responsibility. It should also be stipulated that the technicians have the obligation to foresee the technical code, because the code is highly technical, and the technicians have sufficient knowledge reserves to foresee whether the code may be used for illegal activities, in which case the technicians should also bear the corresponding responsibility.

For the users of smart contracts, they must strictly follow the operating regulations issued by the blockchain smart contract platform, and if the users cause losses to others due to their personal reasons, they should bear the corresponding responsibility for damages and tort liability. However, if the Ether platform fails to strictly manage its employees and creates technical loopholes, which cause actual damages in the end, then the platform should bear the corresponding liability for breach of contract or infringement, and then it has the right to recover the damages from the staff who are at fault.

Because the blockchain smart contract platform involves many subjects in the actual operation, and even there may be hackers to destroy, but these subjects can be regulated within the framework of civil or criminal legal norms, and it is necessary to clarify the responsibility of all the subjects and the way of assuming the responsibility in the legislation, and give full play to the advantages of smart contracts in technology, in order to make it better regulated by the law, which in turn can improve the It is necessary to give full play to the technical advantages of smart contracts in order to make them better regulated by the law, which in turn can improve the operability and applicability of blockchain smart contracts, and constantly broaden the breadth and depth of blockchain smart contracts.

4.3.2. Legal Regulation of Smart Contract Performance under Blockchain:

1) Applying a series of basic principles in the field of civil law to the whole process of performance of blockchain smart contracts. Blockchain smart contract is not completely different from traditional contract, smart contract can be regarded as the result of the integration of computer technology and contract, which still has not jumped out of the essential attribute of contract. Of course, compared with traditional contracts, the formation and fulfillment of blockchain smart contracts are more technical and complex, and the law can't stipulate all the possible types of contracts in reality. Therefore, applying the basic principles of civil law to smart contracts can ensure that legal principles are utilized to resolve smart contract disputes in the absence of legal regulation.

Specifically, the first thing to consider is the principle of autonomy, the private law attributes of market economic relations require that we should first respect the true meaning of the parties to the expression, which is also the essence of private law, smart contracts signed as long as in the case of not violating the mandatory provisions of the law and public order and good morals, the parties to the contract shown in the contract, no validity of the true will of the flaws should be safeguarded, you can independently Decide the object, place and jurisdiction of contracting and other matters, can freely dispose of their rights, increase their obligations or derogate their rights, and should not be interfered by any third party.

Secondly, both parties to the blockchain smart contract still need to abide by the principle of honesty and credit, which is the basis for the conclusion of the contract, and no party can harm the legitimate rights of others for their own selfish desires. When drawing up the contract, both parties should reveal their real will through the form of computer code in the contract, and cannot hide anything from the

other party related to the signing and fulfillment of the contract, or there is fraud, coercion and taking advantage of people's danger and so on, which is a typical violation of the principle of honesty and credit.

Finally, the signing and fulfillment of the blockchain smart contract can not violate the social order and good morals and the green principle, in order to avoid unlawful elements to carry out criminal activities on the smart contract, it is necessary to implement the principles of civil law into the specific implementation of the smart contract, so as to make it adapt to the development and progress of the society.

2) Introducing smart contract-related provisions in civil law. Blockchain smart contracts are contracts in nature, so they should be explicitly incorporated into the Civil Code-Contracts through legislation, and detailed provisions should be made to effectively regulate the behavior of smart contracts.

3) Strict legislative restrictions should be imposed on the subjects, behaviors and responsibilities of the two parties of the blockchain smart contract, so that unless the contract involves others, disinterested third parties are not allowed to participate in the signing and performance of the contract, and the uncertainty risks that may be brought about by the operation of the smart contract should be reduced at the source of the law.

5. Conclusion

This paper implements the construction of a smart contract illegal behavior determination model based on the C4.5 decision tree algorithm, and designs a specific smart contract fulfillment path based on the fulfillment risk assessment of smart contracts.

Among the 24 randomly selected samples, the total prediction probability of the illegal behavior determination model based on C4.5 decision tree algorithm reaches 95.83%, which can effectively identify the illegal behavior of smart contracts.

After 120 iterations of the C4.5 decision tree algorithm, the logarithmic loss function (Logloss) of both the training set and the test set tends to stabilize, and the final model constructed has a better ability to detect illegal behaviors. And among the 24 randomly selected samples, the final measurement results are basically consistent with the real results, and the total prediction probability reaches 95.83%. Further expanding the sample capacity until it reaches the maximum, the final prediction accuracy of the model reaches 93.89%, which reflects that the model in this paper has high prediction ability and applicability to the illegal trading behaviors during the execution of smart contracts.

The Pearson chi-square value between smart contract features and contract risk is 224.6317, and the Sig.(two-tailed) value is 0.000, which indicates that there is a significant correlation between the two, and the asymptotic significance is under 0.001. Meanwhile, from the intersection results of smart contract features and contract risk types, it can be seen that the auto-execution feature and non-tamperable feature of smart contracts lead to the risk of incorrect execution, the transparency feature leads to the risk of code leakage and privacy protection, and the decentralization feature and the cost-benefit feature lead to the regulatory risk. In addition, this paper constructs a contract performance risk evaluation index system based on the hierarchical analysis method, including three first-level indexes of quality risk, schedule risk and funding risk, as well as 10 second-level indexes of production risk, personnel risk, quality management system risk, supporting risk, technical status risk, test quality control risk, production plan risk, quality problem risk, cost control risk and supporting payment risk. Based on the above analysis, this paper finally designs a dynamic monitoring model of smart contract fulfillment risk level, and designs the fulfillment path of smart contract from the aspects of legal responsibility allocation and legal regulation.

References

[1] Gadde, S. S., & Kalli, V. D. (2021). Artificial Intelligence, Smart Contract, and Islamic Finance. International Journal for Research in Applied Science & Engineering Technology (IJRASET), 9(11), 302-310.

[2] Guelida, O., Jai Andaloussi, S., & Ouchetto, O. (2024). Smart Contracts in Finance and Banking Systems in the Era of Industry 5.0: A Systematic Review. Industry 5.0 and Emerging Technologies: Transformation Through Technology and Innovations, 317-346.

[3] Elias, O., Awotunde, O. J., Oladepo, O. I., Azuikpe, P. F., Samson, O. A., Oladele, O. R., & Ogunruku, O. O. (2024). The evolution of green fintech: Leveraging AI and IoT for sustainable financial services and smart contract implementation. World Journal of Advanced Research and Reviews, 23(1), 2710-2723.

[4] Wang, S., Yuan, Y., Wang, X., Li, J., Qin, R., & Wang, F. Y. (2018, June). An overview of smart contract: architecture, applications, and future trends. In 2018 IEEE Intelligent Vehicles Symposium (IV) (pp. 108-113). IEEE.

[5] McNamara, A. J., & Sepasgozar, S. M. (2020). Developing a theoretical framework for intelligent contract acceptance. Construction innovation, 20(3), 421-445.

[6] Yu, F., Bi, W., Cao, N., Li, H., & Higgs, R. (2023). Customer Churn Prediction Framework of Inclusive Finance Based on Blockchain Smart Contract. Computer Systems Science & Engineering, 47(1).

[7] Stathis, G., Trantas, A., Biagioni, G., Graaf, K. A. D., Adriaanse, J., & Herik, J. V. D. (2024). Designing an Intelligent Contract with Communications and Risk Data. SN Computer Science, 5(6), 709.

[8] Rane, N., Choudhary, S., & Rane, J. (2023). Blockchain and Artificial Intelligence (AI) integration for revolutionizing security and transparency in finance. Available at SSRN 4644253.

[9] Wang, S., Ouyang, L., Yuan, Y., Ni, X., Han, X., & Wang, F. Y. (2019). Blockchain-enabled smart contracts: architecture, applications, and future trends. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 49(11), 2266-2277.

[10] Dubey, C., Kumar, D., Singh, A. K., & Dwivedi, V. K. (2022, November). Confluence of Artificial Intelligence and Blockchain Powered Smart Contract in Finance System. In 2022 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS) (pp. 125-130). IEEE.

[11] Mohanta, B. K., Panda, S. S., & Jena, D. (2018, July). An overview of smart contract and use cases in blockchain technology. In 2018 9th international conference on computing, communication and networking technologies (ICCCNT) (pp. 1-4). IEEE.

[12] Taherdoost, H. (2023). Smart contracts in blockchain technology: A critical review. Information, 14(2), 117.

[13] Fauziah, Z., Latifah, H., Omar, X., Khoirunisa, A., & Millah, S. (2020). Application of blockchain technology in smart contracts: A systematic literature review. Aptisi Transactions on Technopreneurship (ATT), 2(2), 160-166.

[14] Balcerzak, A. P., Nica, E., Rogalska, E., Poliak, M., Klieštik, T., & Sabie, O. M. (2022). Blockchain technology and smart contracts in decentralized governance systems. Administrative Sciences, 12(3), 96.

[15] Macrinici, D., Cartofeanu, C., & Gao, S. (2018). Smart contract applications within blockchain technology: A systematic mapping study. Telematics and Informatics, 35(8), 2337-2354.

[16] Yu, J., Qiao, Z., Tang, W., Wang, D., & Cao, X. (2021). Blockchain-based decision tree classification in distributed networks. Intelligent Automation and Soft Computing, 16(8), 10-14.

[17] El Madhoun, N., Hatin, J., & Bertin, E. (2021). A decision tree for building IT applications: What to choose: blockchain or classical systems?. Annals of Telecommunications, 76(3), 131-144.

[18] Fu, M., Zhang, C., Hu, C., Wu, T., Dong, J., & Zhu, L. (2023). Achieving verifiable decision tree prediction on hybrid blockchains. Entropy, 25(7), 1058.

[19] Zhou, Q., Zheng, K., Zhang, K., Hou, L., & Wang, X. (2022). Vulnerability analysis of smart contract for blockchain-based IoT applications: a machine learning approach. IEEE Internet of Things Journal, 9(24), 24695-24707.

[20] Aziz, R. M., Mahto, R., Goel, K., Das, A., Kumar, P., & Saxena, A. (2023). Modified genetic algorithm with deep learning for fraud transactions of ethereum smart contract. Applied Sciences, 13(2), 697.

[21] Ouyang, L., Zhang, W., & Wang, F. Y. (2022). Intelligent contracts: Making smart contracts smart for blockchain intelligence. Computers and Electrical Engineering, 104, 108421.

[22] Sanz Bayón, P. (2019). Key legal issues surrounding smart contract applications. KLRI Journal of Law and Legislation, 9(1), 63-91.

[23] Governatori, G., Idelberger, F., Milosevic, Z., Riveret, R., Sartor, G., & Xu, X. (2018). On legal contracts, imperative and declarative smart contracts, and blockchain systems. Artificial Intelligence and Law, 26, 377-409.

[24] Dolgui, A., Ivanov, D., Potryasaev, S., Sokolov, B., Ivanova, M., & Werner, F. (2020). Blockchain-oriented dynamic modelling of smart contract design and execution in the supply chain. International Journal of Production Research, 58(7), 2184-2199.

[25] Chang, S. E., Chen, Y. C., & Lu, M. F. (2019). Supply chain re-engineering using blockchain technology: A case of smart contract based tracking process. Technological Forecasting and Social Change, 144, 1-11.

[26] Ping Li, Fang Xiong, Xibei Huang & Xiaojun Wen. (2024). Construction and optimization of vending machine decision support system based on improved C4.5 decision tree. Heliyon(3),e25024-.

[27] Hui Xiao, Sha Zeng, Yi Peng & Gang Kou. (2025). A simulation optimization approach for weight valuation in analytic hierarchy process. European Journal of Operational Research(3),851-864.

[28] Ke Chen, Wenshang Chen, Guofu Zou & Ben Chen. (2025). Multi-structure collaborative optimization of porous transport layer for unitized regenerative proton exchange membrane fuel cell based on Taguchi experimental design combined with grey relational analysis. International Journal of Heat and Mass Transfer126471-126471.